

Story:

Delivery boys in e-commerce/ hyperlocal grocery delivery space in India deliver approx. 30-35 orders per day to the customers. For delivery, they have to carry a bunch of printouts containing customer's order details, and after delivery, they have to call their supervisors to report the status of every order. (**Supervisor**, who sits in an office of the e-commerce company, typically handles up to 20-25 delivery boys, and updates the order status in the backend for the orders delivered by those delivery boys). After visiting the customer's premises, they have to call their supervisors to update the delivery status from one of the options mentioned below:

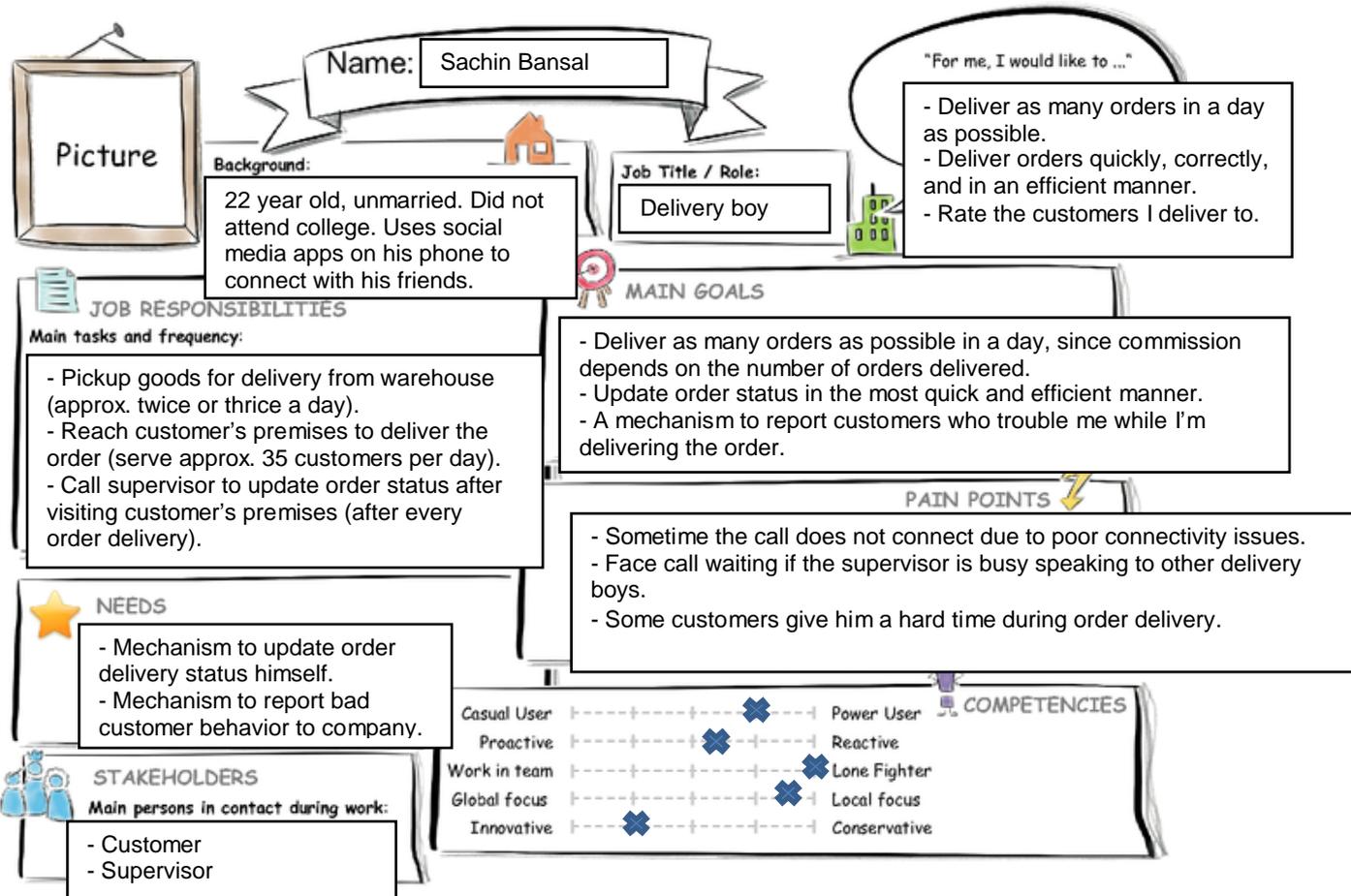
- 1.) Delivered
- 2.) Rejected (due to poor quality of product, for example)
- 3.) Re-schedule (customer not at home, for example)

The "Delivery Status App" aims to remove such dependencies for the delivery boys. They neither need to carry printouts (containing individual order details) anymore, nor make phone calls to their supervisors after visiting the customer location. They can directly check order details and update the delivery status in their Fiori app (offline capability to be added in the app, to mark the order status even in areas of poor network). The e-commerce/logistics company also benefit from this automation as follows:

- Eco-friendly (no individual printouts required)
- They get much more accurate delivery-related metrics/data.
- Removes the dependency on "Supervisors", and they can be involved in some other much more productive work.

A unique feature of this app is that it provides the delivery boys the ability to rate the customers (like Uber drivers can rate their passengers). This can be useful, for example, in the case where the delivery boy did not have a pleasant experience delivering the order to the customer (bad behavior, long waiting time before paying for the order, etc.) In the long run, this data can be useful in flagging/blacklisting customers who repeatedly give delivery boys a hard time.

User Persona:



Point of View (POV):

The **(user)** delivery boy would like to **(need)** mark the order status himself (thus reducing dependency on the Supervisor & avoid network issues). **(Insight)** He would like to have a mechanism to rate the customers he deliver orders to.

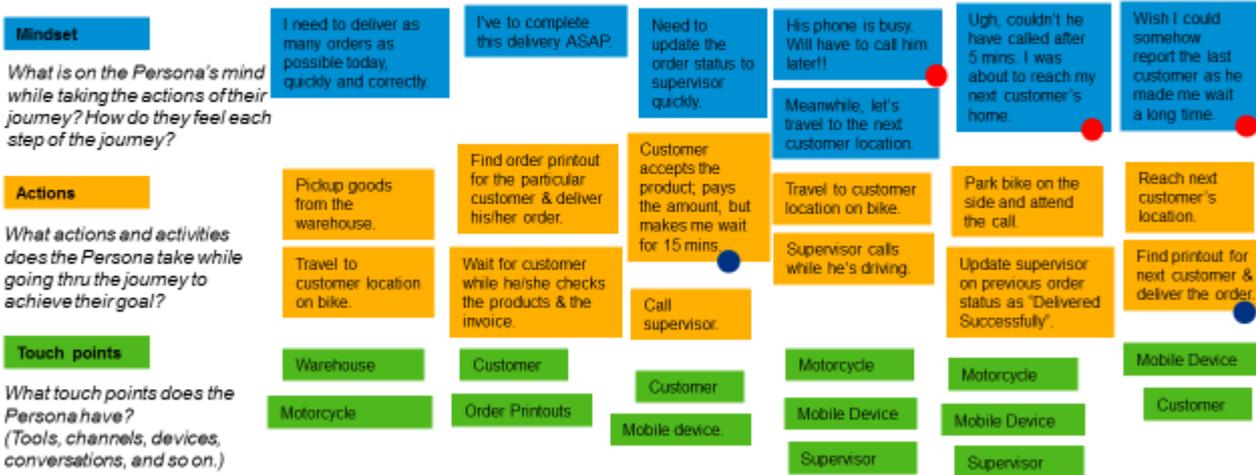
User Experience Journey

Introduction to Design Thinking

User Experience Journey

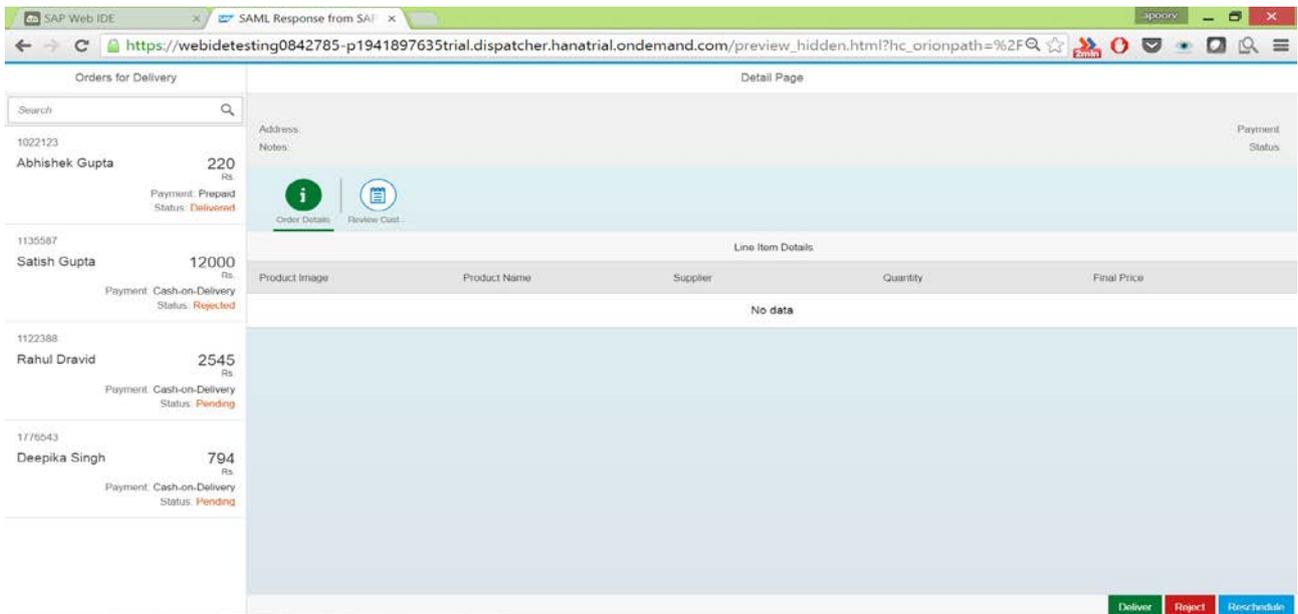
Current User Experience Journey

Duration of the Journey: 20 min



Mock-up (from WebIDE)

- 1.) Application displays all the orders assigned to the delivery boy for the day. Notice the application opens with no list item selected.



2.) Clicking on any particular order (on Master side) will give details about that order (products ordered, price, quantity, etc.) (on the Detail page)

The screenshot shows the SAP Web IDE interface. On the left, the 'Orders for Delivery' list includes:

- 1022123: Abhishek Gupta, 220 Rs, Payment: Prepaid, Status: Delivered
- 1135587: Satish Gupta, 12000 Rs, Payment: Cash-on-Delivery, Status: Rejected
- 1122388: Rahul Dravid, 2545 Rs, Payment: Cash-on-Delivery, Status: Pending
- 1776543: Deepika Singh, 794 Rs, Payment: Cash-on-Delivery, Status: Pending

The 'Detail Page' for order 1022123 shows:

- Customer: Abhishek Gupta
- Address: 41/1 Worli, Mumbai
- Notes: Deliver ASAP
- Amount: 220 Rs
- Payment: Prepaid
- Status: Delivered

The 'Line Item Details' table shows:

Product Image	Product Name	Supplier	Quantity	Final Price
	Flash Drive 4-Gb	Local Hardware Supplier, Chembur, Mumbai	1	220 Rs

(Clicking on 1st customer)

The screenshot shows the SAP Web IDE interface. On the left, the 'Orders for Delivery' list includes:

- 1022123: Abhishek Gupta, 220 Rs, Payment: Prepaid, Status: Delivered
- 1135587: Satish Gupta, 12000 Rs, Payment: Cash-on-Delivery, Status: Rejected
- 1122388: Rahul Dravid, 2545 Rs, Payment: Cash-on-Delivery, Status: Pending
- 1776543: Deepika Singh, 794 Rs, Payment: Cash-on-Delivery, Status: Pending

The 'Detail Page' for order 1122388 shows:

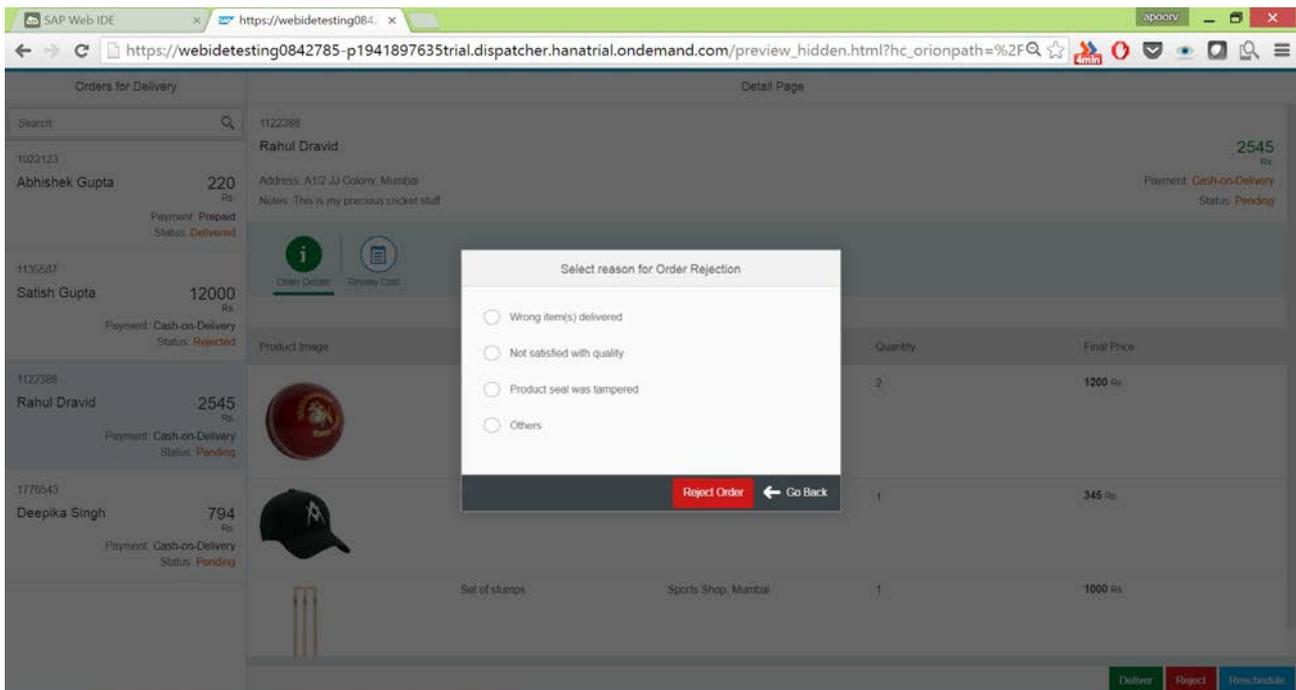
- Customer: Rahul Dravid
- Address: A1/2 JJ Colony, Mumbai
- Notes: This is my precious cricket stuff
- Amount: 2545 Rs
- Payment: Cash-on-Delivery
- Status: Pending

The 'Line Item Details' table shows:

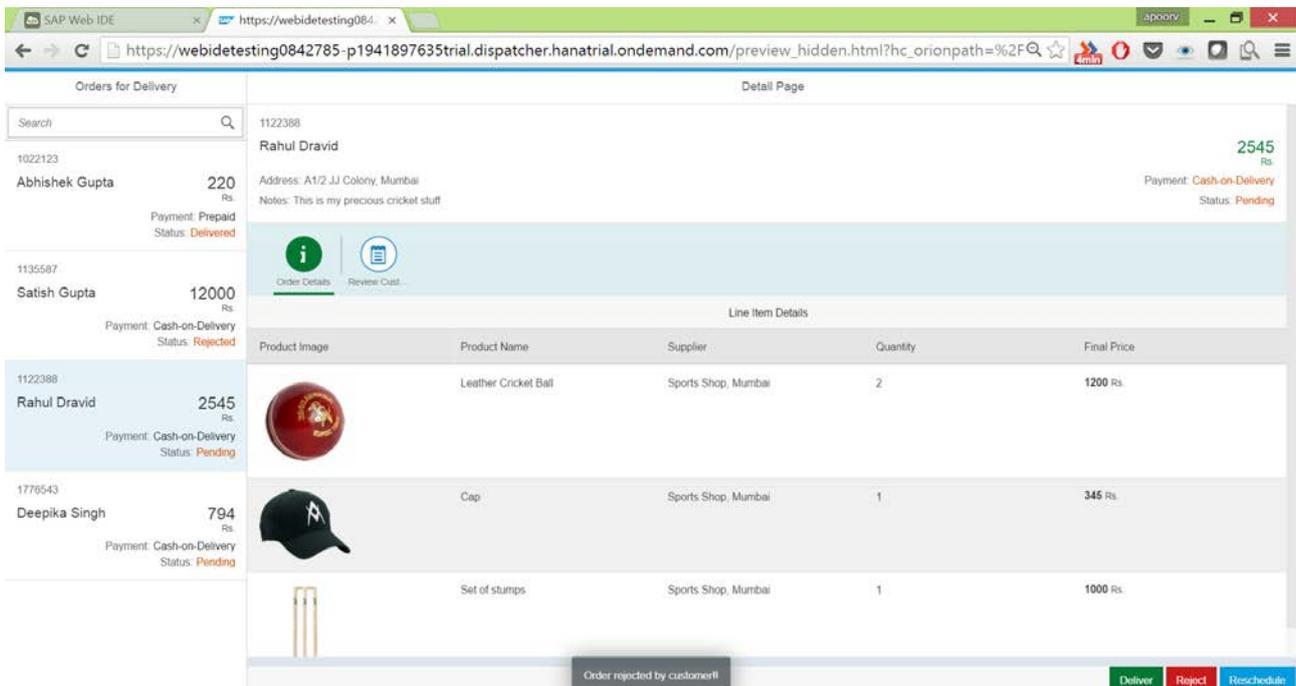
Product Image	Product Name	Supplier	Quantity	Final Price
	Leather Cricket Ball	Sports Shop, Mumbai	2	1200 Rs
	Cap	Sports Shop, Mumbai	1	345 Rs
	Set of stumps	Sports Shop, Mumbai	1	1000 Rs

(Clicking on 3rd customer)

3.) Click on appropriate button in the footer to mark the order status as either Deliver, Reject, or Reschedule.



(Clicking on Reject button for any particular order shows this popup – to select Rejection reason)



(Selecting any reason and clicking “Reject Order” button on the pop-up gives the appropriate message)

4.) Review Customer tab (next to Order Details tab) provides the ability to the delivery boy to rate the customer.

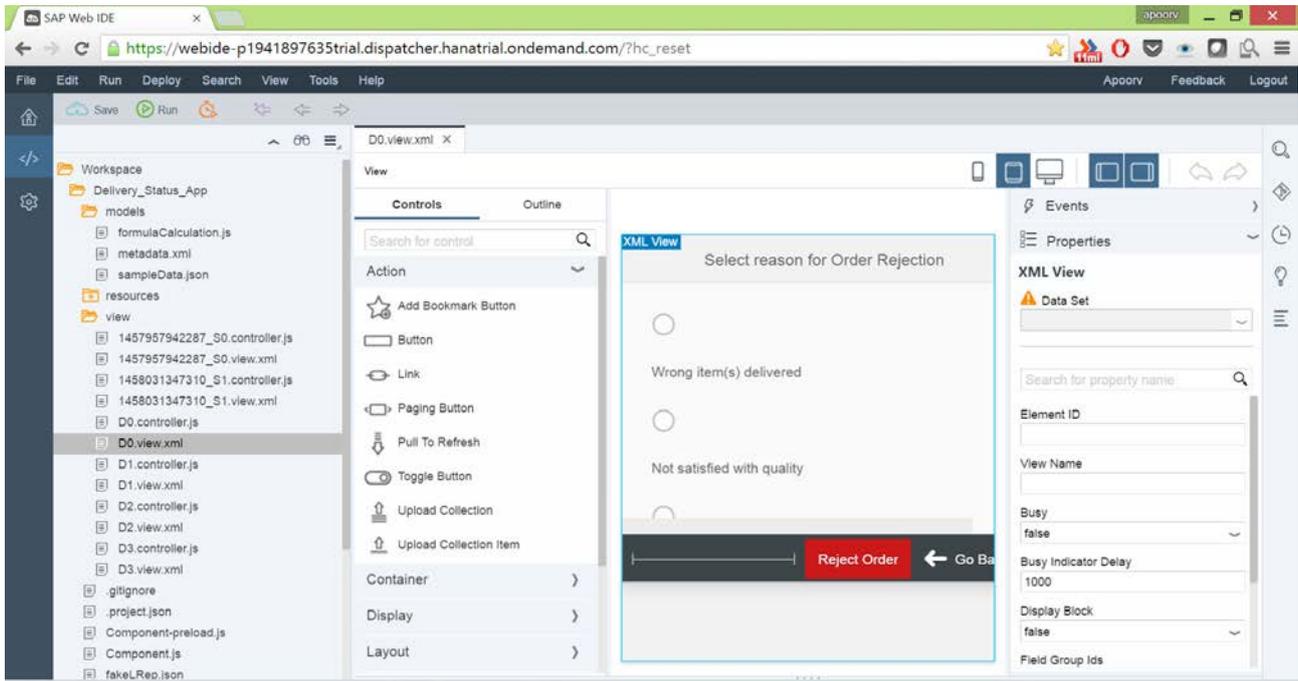
The screenshot displays the SAP Web IDE interface for reviewing a customer. On the left, a list of orders is shown, including Abhishek Gupta (220 Rs, Prepaid, Delivered), Satish Gupta (12000 Rs, Cash-on-Delivery, Rejected), Rahul Dravid (2545 Rs, Cash-on-Delivery, Pending), and Deepika Singh (794 Rs, Cash-on-Delivery, Pending). The main area shows the 'Detail Page' for order 1135587, 'Satish Gupta', with a value of 12000 Rs. The 'Review Customer' section is active, showing a 5-star rating and a 'Submit Feedback' button. Below the rating, there are checkboxes for reasons: 'Long wait-time', 'Payment-related hassles', and 'Bad behavior'. The 'Submit Feedback' button is highlighted in blue.

Rate the customer, assign reasons (if any), and click on Submit Feedback button.

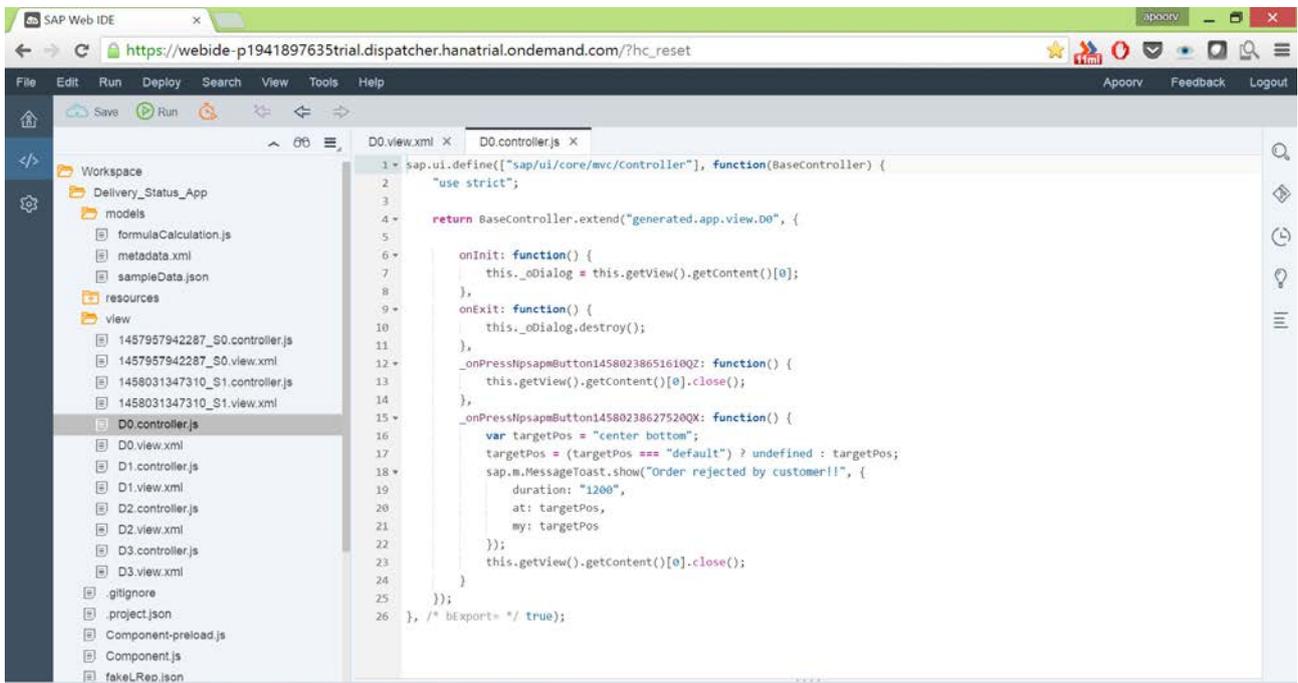
This screenshot shows the same SAP Web IDE interface as the previous one, but with the 'Submit Feedback' button now disabled (greyed out). A dark grey notification box at the bottom center of the screen displays the text 'Customer feedback received'. The 'Review Customer' section still shows the 5-star rating and the reasons for feedback, with 'Payment-related hassles' and 'Bad behavior' checked.

Implementation of 2 Design Components in WebIDE

1.) Pop-up form on click of Reject button on the footer.

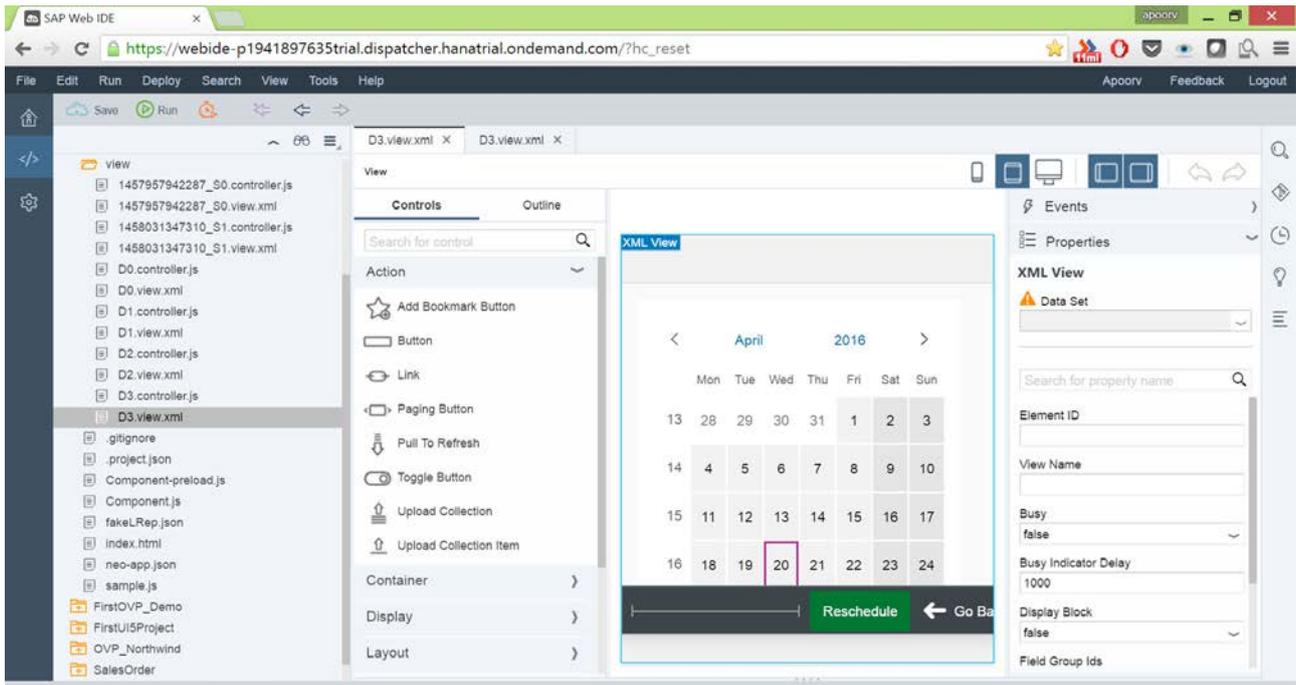


(View created using Layout Editor)

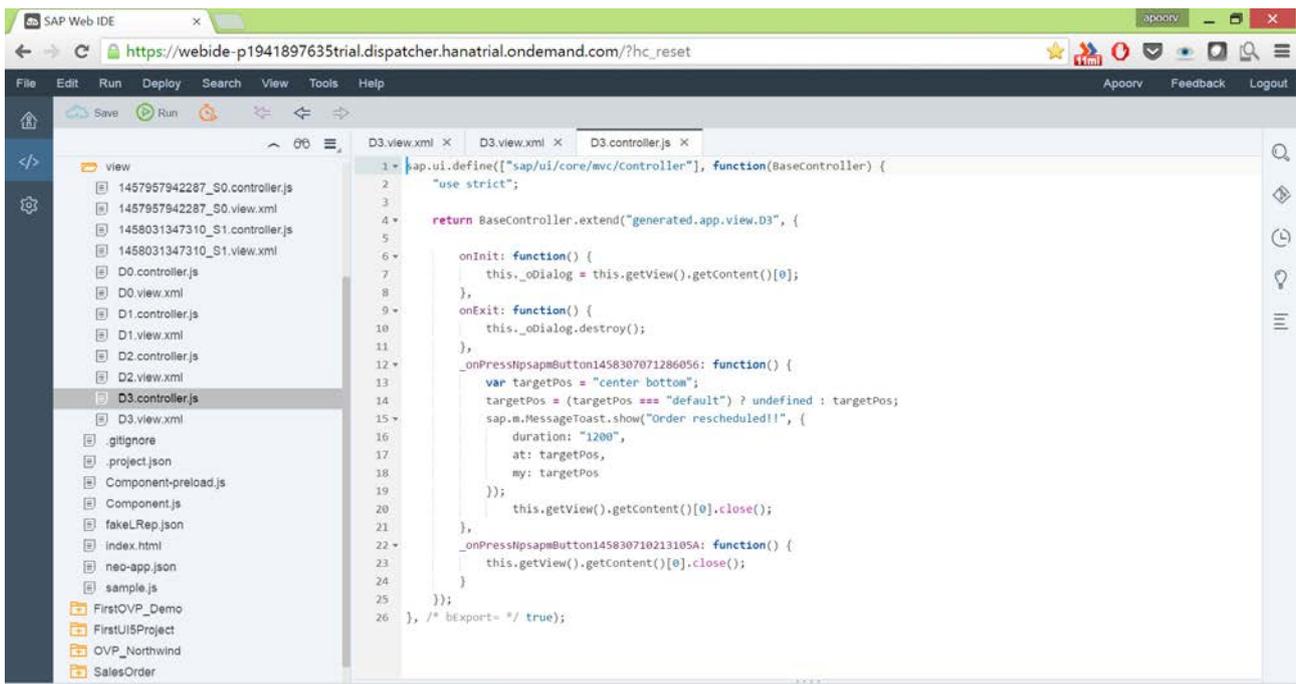


Corresponding controller containing code for button clicks ('Reject Order' and 'Go Back')

2.) Calendar on click of Reschedule button on the footer.



(View created using Layout editor – Upon click of Reschedule button)



Corresponding controller containing code for button clicks ('Reschedule' and 'Go Back')

Study Link

<https://standard.experiencesplash.com/home/projects/f42f2e1ac9be15dc0ba97efc/research/participant/505609bbd9a7dfb10bda1b8a>